
Normalizing Flows for Random Fields in Cosmology

Adam Rouhiainen
Moritz Münchmeyer
Department of Physics
University of Wisconsin-Madison
Madison, WI 53706
rouhiainen@wisc.edu

Utkarsh Giri
Perimeter Institute for Theoretical Physics
Waterloo, ON N2L 2Y5
utkarshgiri18@gmail.com

Abstract

We study the use of normalizing flows to represent the field-level probability density distribution of random fields in cosmology such as the matter and radiation distribution. We evaluate the performance of the real NVP flow for sampling of Gaussian and near-Gaussian random fields, and N-body simulations, and check the quality of samples with different statistics such as power spectrum and bispectrum estimators. We explore aspects of these flows that are specific to cosmology, such as flowing from a physical prior distribution and evaluating the density estimation results in the analytically tractable correlated Gaussian case.

1 Introduction and brief review of normalizing flows

Normalizing flows are a major recent development in probabilistic machine learning [1]. A flurry of recent work has shown their usefulness for a wide range of applications, such as image generation [2, 3], variational inference [4], and likelihood-free inference [5, 6]. In cosmology, normalizing flows have recently been used to represent the posterior distribution of summary statistics such as the power spectrum or cosmological parameters [7, 8]. In the present paper we use normalizing flows to represent the probability density of fields such as the matter distribution directly at field level.

A normalizing flow is a natural way to construct flexible probability distributions by transforming a simple base distribution (often Gaussian) into a complicated target distribution. This is done by applying a series of learned diffeomorphisms to the base distribution. Given a base distribution $p_u(\mathbf{u})$ of a random variable \mathbf{u} , the target distribution $p_x(\mathbf{x})$ is given by $p_x(\mathbf{x}) = p_u(\mathbf{u}) |\det J_T(\mathbf{u})|^{-1}$, where T is the transformation (the “flow”), $\mathbf{x} = T(\mathbf{u})$, and J_T is its Jacobian. We can construct a transformation $T = T_K \circ \dots \circ T_1$ by composing simple transformations T_k , depending on learned parameters and are parametrized using neural networks. In this way very expressive densities can be constructed. Assuming $\mathbf{z}_0 = \mathbf{u}$ and $\mathbf{z}_K = \mathbf{x}$, the transformation at each step k is $\mathbf{z}_k = T_k(\mathbf{z}_{k-1})$ and the Jacobian determinant is $\log |J_T(\mathbf{z})| = \sum_{k=1}^K \log |J_{T_k}(\mathbf{z}_{k-1})|$. Once the flow is learned, two basic statistical operations are performed efficiently: density evaluation and sampling.

While not as expressive as GANs or VAEs, it is a priori plausible that normalizing flows could be particularly strong at representing PDFs of fields in cosmology. Many fields in cosmology, such as the matter distribution, start as Gaussian fields in the far past. They then become progressively more non-Gaussian with time due to non-linear interactions. In the same way, a Gaussian field base distribution of a normalizing flow becomes progressively more non-Gaussian by the applications of more transformations T_k . Such a normalizing flow is therefore a natural candidate to represent somewhat (non-)Gaussian PDFs of matter fields at late times.

In this work, we utilize the real-valued non-volume preserving (real NVP) flow [3]. The real NVP flow is expressive and is fast both for sampling and inference, and has also recently received a lot of attention for its use to represent PDFs in lattice field theory (see e.g. [9]). The details of our network

resemble [9], which we have modified for this work under the CC BY license. We use 3 convolutional layers with kernel size 3 and leaky ReLU activation functions. The number of channels is in this order: 1 (the scalar PDF values), 16 (arbitrary number of feature maps), 16 (arbitrary number of feature maps), 2 (the output variables s and t). As in [9], we stack $K = 16$ affine coupling layers, each with their own CNN to parametrise the affine transformation s and t . We use stride 1 convolutions and no pooling. To implement periodic boundary conditions we use circular padding. To train the flow, we minimize the forward Kullback–Leibler (KL) divergence [1], the relative entropy from the target distribution $p_x^*(\mathbf{x})$ to the base distribution $p_x(\mathbf{x})$. As a function of learned parameters ϕ ,

$$\mathcal{L}(\phi) \approx -\frac{1}{N} \sum_{n=1}^N (\log p_u(T^{-1}(\mathbf{x}_n; \phi)) + \log |\det J_{T^{-1}}(\mathbf{x}_n; \phi)|) + \text{const.}, \quad (1)$$

where T is the flow transformation. We use the Adam optimizer to minimize the loss with respect to the parameters ϕ , with a learning rate of 0.001. We also normalize the training samples to unit variance and use a batch size of 128. Our results are reproducible in Jupyter Notebooks available at github.com/SubmissionForPapers/NormalizingFlowsCosmology.

2 Application to Gaussian fields with density evaluation

We train our real NVP flow on samples from a correlated Gaussian field, with a CMB temperature power spectrum. Here we flow from an uncorrelated Gaussian prior distribution. We use patches of 64^2 pixels covering a sky angle side length of 4 degrees. In this first example, we use infinite *on the fly* created training samples. Training time was about 40 hours on an RTX 3090 and used about 3.8 GB of GPU memory. Samples from the prior, model, and target (training) distribution are shown in Fig. 1 (left). By eye, the model samples look like the training data.

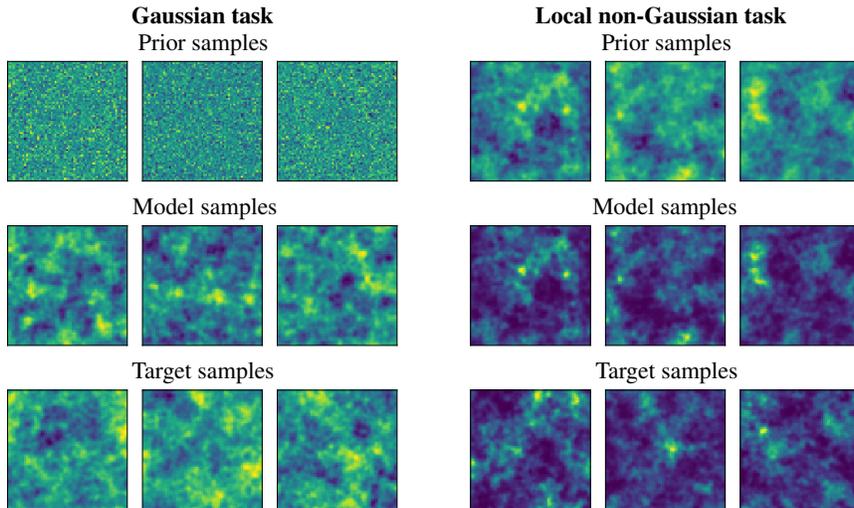


Figure 1: Prior samples (top), flow samples (middle), and training samples (bottom). Left: Gaussian field samples. Right: Local non-Gaussian field samples with $f_{NL}^{\text{local}} = 0.2$. The non-Gaussian training data is generated by drawing Gaussian maps from a CMB power spectrum on a 4 degree sky patch represented on 64^2 pixels and making them non-Gaussian with Eq. 2. The network makes the correlated Gaussian prior samples more non-Gaussian, with extrema becoming more pronounced.

For some of the data analysis applications discussed in the introduction, we need to use the flow in reverse direction for density evaluation. There are two different tasks we can consider here: in distribution density evaluation (IID: independent, identical distribution) and out of distribution (OOD) density evaluation; we focus here on IID density evaluation. We would like to verify that IID samples, when run backwards through the flow (with uncorrelated Gaussian noise prior), are assigned a probability that corresponds to their true probability. Here the fact that we start with a tractable Gaussian distribution allows us to compare the flow probability with the exact analytic probability. We sample 10,000 new IID samples x from the same distribution as the training data

Table 1: Local non-Gaussianity measurements

$\tilde{f}_{NL}^{\text{local}}$ training	$\tilde{f}_{NL}^{\text{local}}$ flow	$\tilde{f}_{NL}^{\text{equi}}$ training	$\tilde{f}_{NL}^{\text{equi}}$ flow
0.2 ± 0.074	0.187 ± 0.085	0.150 ± 0.044	0.146 ± 0.040
0.05 ± 0.024	0.047 ± 0.029	0.037 ± 0.017	0.035 ± 0.017

and reverse flow them to obtain $\log P^{\text{flow}}(x)$. For the configuration above, we find that the cross correlation coefficient between $\log P^{\text{flow}}(x)$ and $\log P^{\text{true}}(x)$ is $r \simeq 0.993$. We plot these quantities for 200 random example maps in Fig. 2. When we limit training data to 1,000 Gaussian maps, the cross correlation coefficient dropped to about $r \simeq 0.97$, while with 10,000 training maps we found $r \simeq 0.98$. The flow thus learns density evaluation on IID samples rather well.

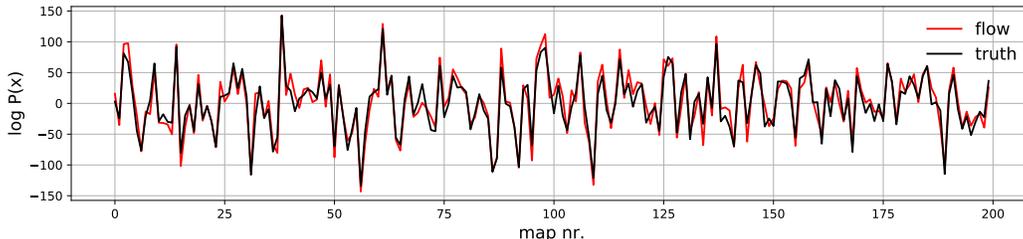


Figure 2: Density evaluation $\log P(x)$ result from the flow compared to the true log probability of the sample (mean shifted to zero), for 200 maps with 64^2 pixel resolution. The cross-correlation coefficient is 0.993, estimated from 10,000 maps.

3 Local non-Gaussianity and correlated prior

We consider the simplest non-Gaussian random field in cosmology, the *local non-Gaussianity*. Local non-Gaussianity [10] is generated by transforming a correlated Gaussian field $\phi_G(\mathbf{x})$ as

$$\phi_{NG}(\mathbf{x}) = \phi_G(\mathbf{x}) + \tilde{f}_{NL}^{\text{local}} (\phi_G(\mathbf{x})^2 - \langle \phi_G(\mathbf{x})^2 \rangle) \quad (2)$$

To draw samples from this distribution, we square the Gaussian field and add it to the original field with some amplitude. This form of non-Gaussianity is generated in cosmology for example by multi-field inflation [11]. We normalize ϕ_G to variance 1 so that $\tilde{f}_{NL}^{\text{local}} = 1$ indicates $\mathcal{O}(1)$ non-Gaussianity. This time we flow from a correlated Gaussian field with the correct power spectrum. This makes the training much more efficient as the flow has a better starting point. We show samples from the prior, model, and training distribution in Fig. 1 (right). The samples are indistinguishable by eye from the training data. We show the power spectrum of the samples, prior, and training data in Fig. 3 (left). The flow thus learns to induce the right non-Gaussianity while keeping the power spectrum intact. The correlated prior also improves the training convergence. We measure the non-Gaussianity in the samples by estimating the amplitude of two non-Gaussian templates, the local and equilateral non-Gaussianity. In the present case the training data has local non-Gaussianity by definition, however the equilateral template has some overlap with the local template, and it is useful to measure both in the model samples whose non-Gaussianity can be non-local. We record mean estimated values and per sample variance of the estimated non-Gaussianity in Table 1, averaging over 10,000 training and flow samples. The flow is accurate to about 5% in the mean and 10% in the per sample variance.

4 Non-Gaussian fields from N-body simulations

Here we tackle the important use case of representing field PDFs from N-body simulations. We use 100 high-resolution simulations from the Quijote [12] suite of N-body simulation data (provided under the MIT License) to generate training patches of the 2D matter field. The simulations were run in a box size of $1 h^{-1}$ Gpc and use 1024^3 dark matter particles. We use snapshots generated at $z = 2$ to estimate the matter density by painting the matter particles on a 3D mesh of size 1024^3 using the Cloud-in-Cell [13] mass assignment scheme implemented in `nbodykit` [14]. We are able to resolve

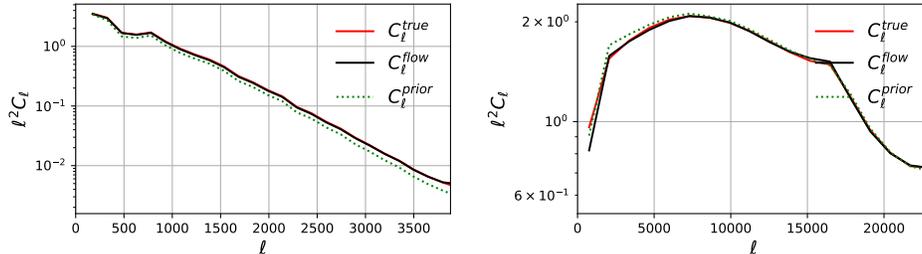


Figure 3: Left: Comparison of the prior, model, and training set power spectrum averaged over 10,000 maps, for the $f_{NL} = 0.2$ example. With the correlated prior, the BAO wiggle is resolved perfectly. Right: Projected matter power spectrum of the N-body simulation samples (red), trained flow model (black) and correlated input prior (green), averaging over 10,000 maps.

Table 2: N-body non-Gaussianity measurements

$\tilde{f}_{NL}^{\text{local}}$ training	$\tilde{f}_{NL}^{\text{local}}$ flow	$\tilde{f}_{NL}^{\text{equi}}$ training	$\tilde{f}_{NL}^{\text{equi}}$ flow
0.85 ± 0.041	0.78 ± 0.29	0.61 ± 0.24	0.62 ± 0.24

nonlinear scales of size $k \gtrsim 1 h^{-1}$ Mpc. We divide the simulation volume into boxes of size 128^3 and project the density to get a realization of a 2D field. We generated 51,200 independent patches. Limited by GPU memory, larger maps would require a method to patch these 128^3 maps together.

We train the same real NVP flow as the previous sections. Training time was 50 hours on an RTX 3090, using 11 GB of memory. We found faster convergence and better sample quality from a correlated Gaussian prior with the right power spectrum compared to flowing from Gaussian noise. In Fig. 3 (right) we find a near perfect power spectrum of the flow. Non-Gaussianity measurements are recorded in Table 2, sampling over 10,000 training and model maps. The flow is accurate to 5 to 10%.

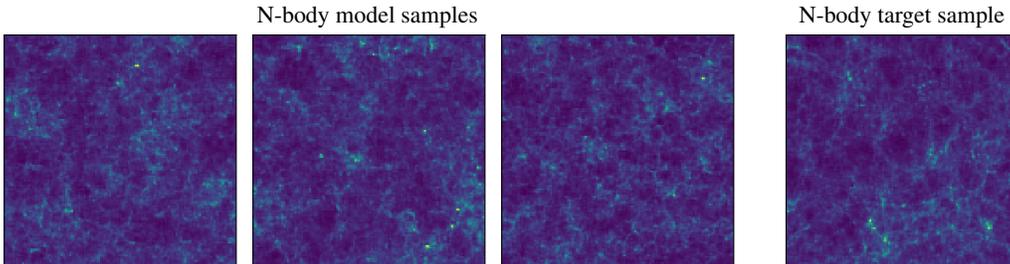


Figure 4: N-body simulation samples trained on real NVP flow. Each sample is derived from a 3D volume of $(125 h^{-1} \text{ Mpc})^3$ by projecting the matter field such that the 2D map has a resolution of 128^2 pixels. Left: Model samples from the trained flow. Right: Sample from the training data.

5 Conclusion

The goal of this study is to evaluate how good normalizing flows are at representing PDFs of random fields in cosmology, both for sampling and density evaluation. We find that we can train flows whose samples have power spectra and non-Gaussianity accurate to a few percent. We tested the inverse operation of density evaluation in the analytically tractable Gaussian case and found percent level accuracy in $\log P$. Using the non-Gaussianity estimation formalism of cosmology [15], we compared the non-Gaussianity measurement of the flow distribution to the target distribution. Measuring three-point correlation functions in statistical machine learning may be generally useful to quantify the performance of normalizing flows in representing PDFs. In upcoming work we will apply these flows to inference problems in cosmology. For example, analysing a small but deep sky patch with HMC such as for CMB lensing in [16] would benefit from a non-Gaussian prior distribution of the lensing

potential represented with a normalizing flow. Normalizing flows are also suitable for variational inference of the initial conditions of the matter distribution of the universe. For some applications it would be useful to make the flow PDF conditionally dependent on astrophysical or cosmological parameters. Memory constraints will require a patching scheme to represent larger maps, or 3D maps.

References

- [1] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing Flows for Probabilistic Modeling and Inference. *arXiv e-prints*, *arXiv:1912.02762*, December 2019.
- [2] Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative Flow with Invertible 1x1 Convolutions. *arXiv e-prints*, *arXiv:1807.03039*, July 2018.
- [3] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using Real NVP. *arXiv e-prints*, *arXiv:1605.08803*, May 2016.
- [4] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv e-prints*, *arXiv:1505.05770*, June 2016.
- [5] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked Autoregressive Flow for Density Estimation. *arXiv e-prints*, *arXiv:1705.07057*, May 2017.
- [6] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *arXiv e-prints*, *arXiv:1911.01429*, November 2019.
- [7] Justin Alsing, Tom Charnock, Stephen Feeney, and Benjamin Wandelt. Fast likelihood-free cosmology with neural density estimators and active learning. *Mon. Not. Roy. Astron. Soc.*, 488(3):4440–4458, February 2019.
- [8] Ana Diaz Rivero and Cora Dvorkin. Flow-Based Likelihoods for Non-Gaussian Inference. *Phys. Rev. D*, 102(10):103507, July 2020.
- [9] Michael S. Alberg, Denis Boyda, Daniel C. Hackett, Gurtej Kanwar, Kyle Cranmer, Sébastien Racanière, Danilo Jimenez Rezende, and Phiala E. Shanahan. Introduction to Normalizing Flows for Lattice Field Theory. *arXiv e-prints*, *arXiv:2101.08176*, January 2021.
- [10] Eiichiro Komatsu and David N. Spergel. Acoustic signatures in the primary microwave background bispectrum. *Phys. Rev. D*, 63:063002, February 2001.
- [11] Christian T. Byrnes and Ki-Young Choi. Review of local non-Gaussianity from multi-field inflation. *Adv. Astron.*, 2010:724525, February 2010.
- [12] Francisco Villaescusa-Navarro, ChangHoon Hahn, Elena Massara, Arka Banerjee, Ana Maria Delgado, Doogesh Kodi Ramanah, Tom Charnock, Elena Giusarma, Yin Li, Erwan Allys, and et al. The Quijote simulations. *The Astrophysical Journal Supplement Series*, 250(1):2, August 2020.
- [13] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. 1981.
- [14] Nick Hand, Yu Feng, Florian Beutler, Yin Li, Chirag Modi, Uros Seljak, and Zachary Slepian. nbodykit: an open-source, massively parallel toolkit for large-scale structure. *Astron. J.*, 156(4):160, September 2018.
- [15] J. R. Fergusson, D. M. Regan, and E. P. S. Shellard. Rapid Separable Analysis of Higher Order Correlators in Large Scale Structure. *Phys. Rev. D*, 86:063511, September 2012.
- [16] M. Millea et al. Optimal CMB Lensing Reconstruction and Parameter Estimation with SPTpol Data. *arXiv e-prints*, *arXiv:2012.01709*, December 2020.

Checklist

1. For all authors...
 - (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? **[Yes]**
 - (b) Did you describe the limitations of your work? **[Yes]** See Section 1, Section 4, and Section 5.
 - (c) Did you discuss any potential negative societal impacts of your work? **[No]** We modify an existing generative model for cosmological applications.
 - (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? **[Yes]**
2. If you are including theoretical results...
 - (a) Did you state the full set of assumptions of all theoretical results? **[N/A]**
 - (b) Did you include complete proofs of all theoretical results? **[N/A]**
3. If you ran experiments...
 - (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? **[Yes]** See Section 1.
 - (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? **[Yes]** See Section 1.
 - (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? **[Yes]** See Table 1 and Table 2.
 - (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? **[Yes]** See Section 2 and Section 4.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
 - (a) If your work uses existing assets, did you cite the creators? **[Yes]** See Section 1 for the real NVP code and Section 4 for the Quijote data.
 - (b) Did you mention the license of the assets? **[Yes]** See Section 1 for the real NVP model and Section 4 for the Quijote data.
 - (c) Did you include any new assets either in the supplemental material or as a URL? **[Yes]** See Section 1.
 - (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? **[No]** The Quijote data was obtained from physical simulations. The licenses for the real NVP code and Quijote data give us permission, and we used them for their intended purposes.
 - (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? **[No]** We don't believe it does; we use cosmological data from simulations.
5. If you used crowdsourcing or conducted research with human subjects...
 - (a) Did you include the full text of instructions given to participants and screenshots, if applicable? **[N/A]**
 - (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? **[N/A]**
 - (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? **[N/A]**